Large-Scale Machine Learning and Applications

Soutenance pour l'habilitation à diriger des recherches

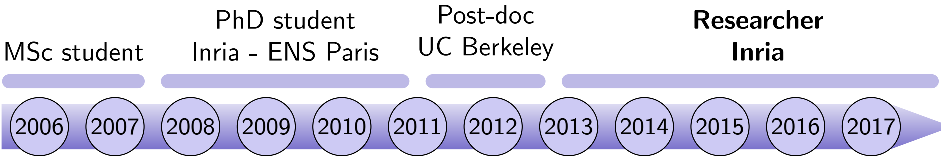
Julien Mairal

Jury:

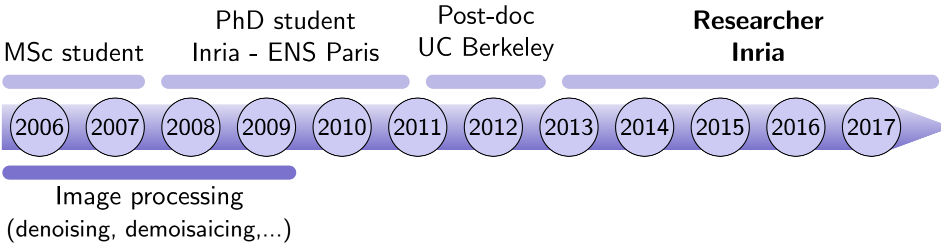
Pr. Léon Bottou	NYU & Facebook	Rapporteur
Pr. Mário Figueiredo	IST, Univ. Lisbon	Examinateur
Dr. Yves Grandvalet	CNRS	Rapporteur
Pr. Anatoli Judistky	Univ. Grenoble-Alpes	Examinateur
Pr. Klaus-Robert Müller	TU Berlin	Rapporteur
Dr. Florent Perronnin	Naver Labs	Examinateur
Dr. Cordelia Schmid	Inria	Examinateur

Part I: Introduction

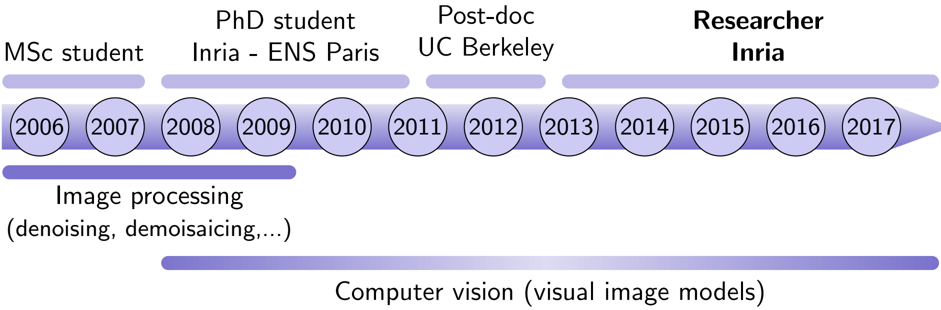
Short overview of my past research



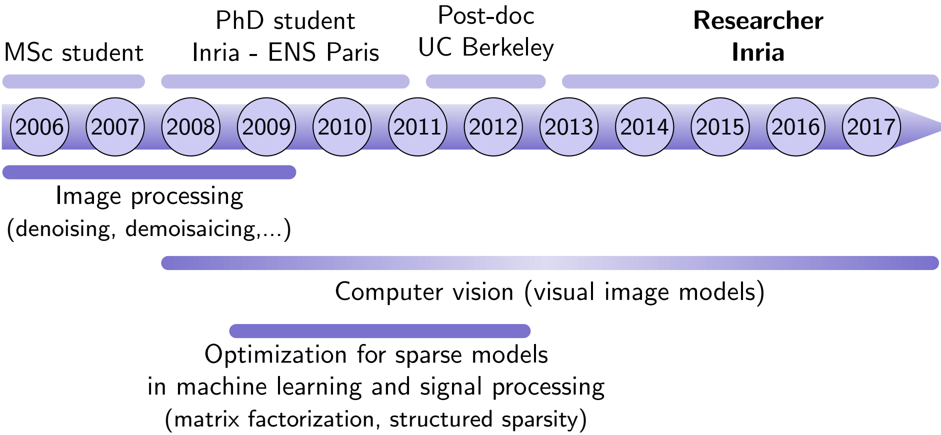
Short overview of my past research



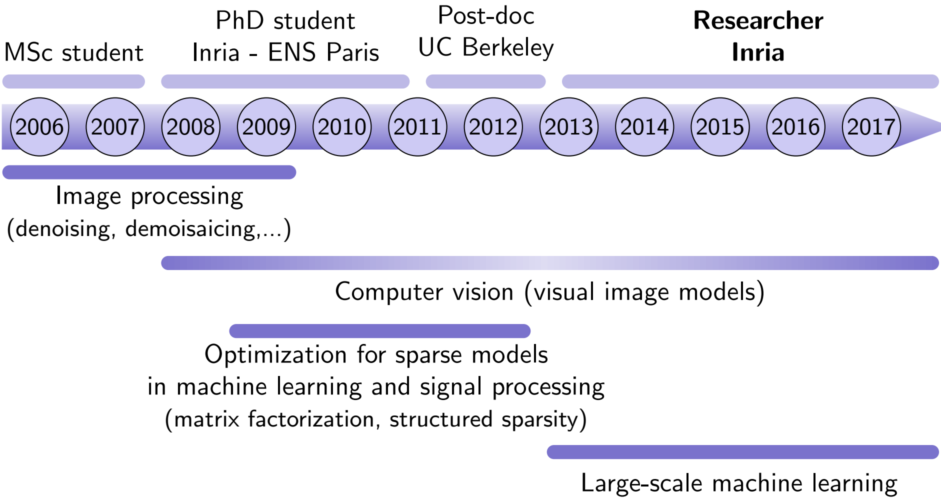
Short overview of my past research



Short overview of my past research



Short overview of my past research



Short overview of my past research

MSc student

PhD student
Inria - ENS Paris

Post-doc
UC Berkeley

Researcher
Inria

2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017

Image processing
(denoising, demoisaicing,...)

Pluri-disciplinary collaborations
(bioinformatics, neuroimaging, neuroscience)

Computer vision (visual image models)

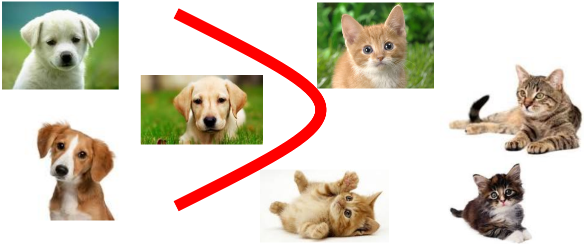
Optimization for sparse models
in machine learning and signal processing
(matrix factorization, structured sparsity)

Large-scale machine learning

Paradigm 1: Machine learning as optimization problems

Optimization is central to machine learning. For instance, in supervised learning, the goal is to learn a **prediction function** $f : \mathcal{X} \to \mathcal{Y}$ given labeled training data $(x_i, y_i)_{i=1,\dots,n}$ with x_i in $\mathcal{X}$, and y_i in $\mathcal{Y}$:

$$\min_{f \in \mathcal{F}} \underbrace{\frac{1}{n}\sum_{i=1}^{n} L(y_i, f(x_i))}_{\text{empirical risk, data fit}} + \underbrace{\lambda\Omega(f)}_{\text{regularization}}.$$



[Vapnik, 1995, Bottou, Curtis, and Nocedal, 2016]...

Paradigm 1: Machine learning as optimization problems

Optimization is central to machine learning. For instance, in supervised learning, the goal is to learn a **prediction function** $f : \mathcal{X} \to \mathcal{Y}$ given labeled training data $(x_i, y_i)_{i=1,\ldots,n}$ with x_i in $\mathcal{X}$, and y_i in $\mathcal{Y}$:

$$\min_{f \in \mathcal{F}} \underbrace{\frac{1}{n}\sum_{i=1}^{n} L(y_i, f(x_i))}_{\text{empirical risk, data fit}} + \underbrace{\lambda \Omega(f)}_{\text{regularization}} .$$

Example with linear models: logistic regression, SVMs, *etc.*

- $f(x) = w^\top x + b$ is parametrized by w, b in $\mathbb{R}^{p+1}$;
- L is a **convex** loss function;
- …but n and p may be **huge** $\geq 10^6$.

Paradigm 1: Machine learning as optimization problems

Optimization is central to machine learning. For instance, in supervised learning, the goal is to learn a **prediction function** $f : \mathcal{X} \to \mathcal{Y}$ given labeled training data $(x_i, y_i)_{i=1,\ldots,n}$ with x_i in $\mathcal{X}$, and y_i in $\mathcal{Y}$:

$$\min_{f \in \mathcal{F}} \underbrace{\frac{1}{n}\sum_{i=1}^{n} L(y_i, f(x_i))}_{\text{empirical risk, data fit}} + \underbrace{\lambda\Omega(f)}_{\text{regularization}} .$$

Example with deep learning

- The "deep learning" space $\mathcal{F}$ is parametrized:

$$f(x) = \sigma_k(A_k\sigma_{k-1}(A_{k-1}\ldots\sigma_2(A_2\sigma_1(A_1x))\ldots)).$$

- Finding the optimal $A_1, A_2, \ldots, A_k$ yields an (intractable) **non-convex** optimization problem in **huge dimension.**

Paradigm 1: Machine learning as optimization problems

Optimization is central to machine learning. For instance, in supervised learning, the goal is to learn a **prediction function** $f : \mathcal{X} \to \mathcal{Y}$ given labeled training data $(x_i, y_i)_{i=1,\ldots,n}$ with x_i in $\mathcal{X}$, and y_i in $\mathcal{Y}$:

$$\min_{f \in \mathcal{F}} \underbrace{\frac{1}{n}\sum_{i=1}^{n} L(y_i, f(x_i))}_{\text{empirical risk, data fit}} + \underbrace{\lambda\Omega(f)}_{\text{regularization}} .$$

Today's challenges: develop algorithms that

- **scale** both in the problem size n and dimension p;
- are able to **exploit the problem structure** (sum, composite);
- come with **convergence and numerical stability** guarantees;
- come with **statistical guarantees**.

Paradigm 2: The sparsity principle

The way we do machine learning follows a classical scientific paradigm:

1. **observe** the world (gather data);
2. **propose models** of the world (design and learn);
3. **test** on new data (estimate the generalization error).

[Corfield et al., 2009].

Paradigm 2: The sparsity principle

The way we do machine learning follows a classical scientific paradigm:

1. **observe** the world (gather data);
2. **propose models** of the world (design and learn);
3. **test** on new data (estimate the generalization error).

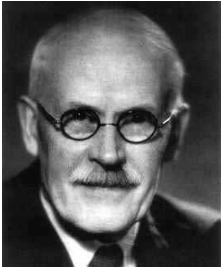
But...

- it is not always possible to distinguish the generalization error of various models based on available data.
- when a complex model A performs slightly better than a simple model B, should we prefer A or B?
- generalization error requires a predictive task: what about unsupervised learning? which measure should we use?
- we are also leaving aside the problem of non i.i.d. train/test data, biased data, testing with counterfactual reasoning...

[Corfield et al., 2009, Bottou et al., 2013, Schölkopf et al., 2012].

Paradigm 2: The sparsity principle

(a) Dorothy Wrinch
1894–1980



(b) Harold Jeffreys
1891–1989

The existence of simple laws is, then, apparently, to be regarded as a quality of nature; and accordingly we may infer that it is justifiable to prefer a simple law to a more complex one that fits our observations slightly better.

[Wrinch and Jeffreys, 1921].

Paradigm 2: The sparsity principle

Remarks: sparsity is...

- appealing for experimental sciences for **model interpretation**;
- (too-)**well understood** in some mathematical contexts:

$$\min_{w \in \mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n} L\left(y_i, w^\top x_i\right)}_{\text{empirical risk, data fit}} + \underbrace{\lambda \|w\|_1}_{\text{regularization}}.$$

- extremely powerful for **unsupervised learning** in the context of matrix factorization, and **simple to use**.

[Olshausen and Field, 1996, Chen, Donoho, and Saunders, 1999, Tibshirani, 1996]...

Paradigm 2: The sparsity principle

Remarks: sparsity is...

- appealing for experimental sciences for **model interpretation**;
- (too-)**well understood** in some mathematical contexts:

$$\min_{w \in \mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n} L\left(y_i, w^\top x_i\right)}_{\text{empirical risk, data fit}} + \underbrace{\lambda \|w\|_1}_{\text{regularization}} .$$

- extremely powerful for **unsupervised learning** in the context of matrix factorization, and **simple to use**.

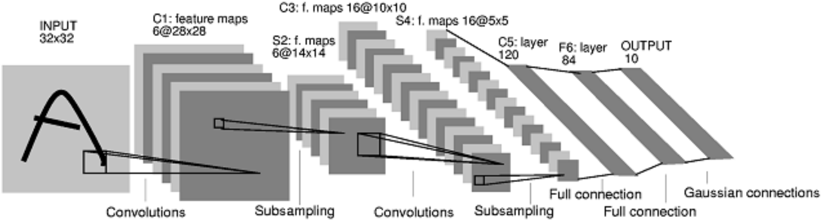
Today's challenges

- Develop sparse and **stable** (and **invariant?**) models.
- Go beyond clustering / low-rank / union of subspaces.

[Olshausen and Field, 1996, Chen, Donoho, and Saunders, 1999, Tibshirani, 1996]...

Paradigm 3: Deep Kernel Machines

A quick zoom on convolutional neural networks



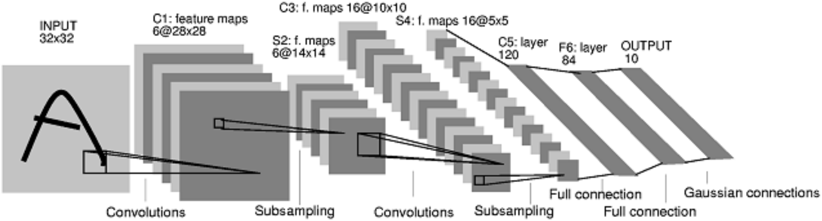
- still involves the ERM problem

$$\min_{f \in \mathcal{F}} \underbrace{\frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i))}_{\text{empirical risk, data fit}} + \underbrace{\lambda \Omega(f)}_{\text{regularization}}.$$

[LeCun et al., 1989, 1998, Ciresan et al., 2012, Krizhevsky et al., 2012]...

Paradigm 3: Deep Kernel Machines

A quick zoom on convolutional neural networks

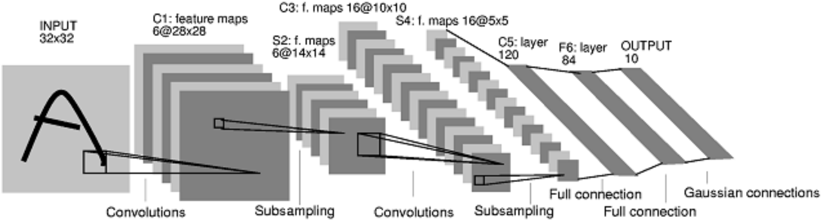


What are the main features of CNNs?

- they capture **compositional** and **multiscale** structures in images;
- they provide some **invariance**;
- they model **local stationarity** of images at several scales.

Paradigm 3: Deep Kernel Machines

A quick zoom on convolutional neural networks



What are the main open problems?

- very little **theoretical understanding**;
- they require **large amounts of labeled data**;
- they require **manual design and parameter tuning**;

Paradigm 3: Deep Kernel Machines

A quick zoom on kernel methods

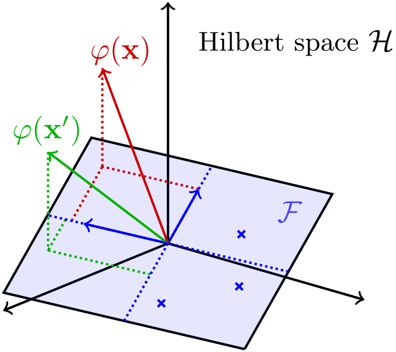
1. map data to a Hilbert space:

$$\varphi : \mathcal{X} \rightarrow \mathcal{H}.$$

2. work with linear forms f in $\mathcal{H}$:

$$f(x) = \langle \varphi(x), f \rangle_{\mathcal{H}}.$$

3. run your favorite algorithm in $\mathcal{H}$ (PCA, CCA, SVM, ...)



- all we need is a **positive definite kernel** function $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$

$$K(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}.$$

Paradigm 3: Deep Kernel Machines

A quick zoom on kernel methods

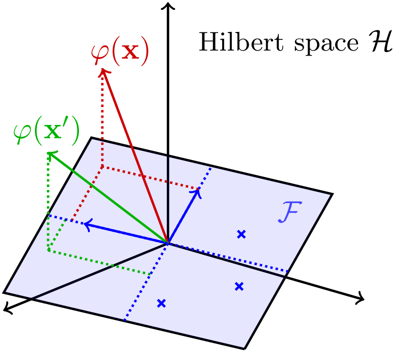
1. map data to a Hilbert space:

$$\varphi : \mathcal{X} \rightarrow \mathcal{H}.$$

2. work with linear forms f in $\mathcal{H}$:

$$f(x) = \langle \varphi(x), f \rangle_{\mathcal{H}}.$$

3. run your favorite algorithm in $\mathcal{H}$ (PCA, CCA, SVM, ...)



- for supervised learning, it also yields the ERM problem

$$\min_{f \in \mathcal{H}} \; \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) \; + \; \lambda \|f\|^2_{\mathcal{H}}.$$

Paradigm 3: Deep Kernel Machines

What are the main features of kernel methods?

- builds **well-studied functional spaces** to do machine learning;
- **decoupling** of data representation and learning algorithm;
- typically, **convex optimization problems** in a supervised context;
- **versatility**: applies to vectors, sequences, graphs, sets,. . . ;
- **natural regularization function** to control the learning capacity;

[Shawe-Taylor and Cristianini, 2004, Schölkopf and Smola, 2002, Müller et al., 2001]

Paradigm 3: Deep Kernel Machines

What are the main features of kernel methods?

- builds **well-studied functional spaces** to do machine learning;
- **decoupling** of data representation and learning algorithm;
- typically, **convex optimization problems** in a supervised context;
- **versatility**: applies to vectors, sequences, graphs, sets, . . . ;
- **natural regularization function** to control the learning capacity;

But...

- **decoupling** of data representation and learning may not be a good thing, according to recent **supervised** deep learning success.
- requires **kernel design**.

[Shawe-Taylor and Cristianini, 2004, Schölkopf and Smola, 2002, Müller et al., 2001]

Paradigm 3: Deep Kernel Machines

Challenges of deep kernel machines

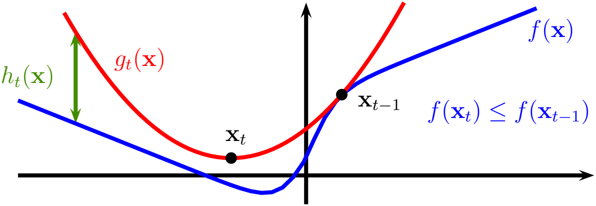
- **Build functional spaces for deep learning**, where we can quantify **invariance and stability** to perturbations, **signal recovery** properties, and the **complexity** of the function class.
- do deep learning with a **geometrical interpretation** (learn collections of linear subspaces, perform projections).
- exploit kernels for **structured objects** (graph, sequences) within deep architectures.
- show that **end-to-end** learning is natural with kernel methods.

Part II: Contributions

Contributions of this HdR

Axis 1: large-scale optimization for machine learning

- Structured MM algorithms for structured problems.



Contributions of this HdR

Axis 1: large-scale optimization for machine learning

- Structured MM algorithms for structured problems.
- Variance-reduced stochastic optimization for convex optimization.

Contributions of this HdR

Axis 1: large-scale optimization for machine learning

- Structured MM algorithms for structured problems.
- Variance-reduced stochastic optimization for convex optimization.
- **Acceleration by smoothing.**

Contributions of this HdR

Axis 1: large-scale optimization for machine learning

- Structured MM algorithms for structured problems.
- Variance-reduced stochastic optimization for convex optimization.
- **Acceleration by smoothing.**

Axis 2: Deep kernel machines

- **Convolutional kernel networks.**

Contributions of this HdR

Axis 1: large-scale optimization for machine learning

- Structured MM algorithms for structured problems.
- Variance-reduced stochastic optimization for convex optimization.
- **Acceleration by smoothing.**

Axis 2: Deep kernel machines

- **Convolutional kernel networks.**
- Applications to image retrieval and image super-resolution.

Contributions of this HdR

Axis 1: large-scale optimization for machine learning

- Structured MM algorithms for structured problems.
- Variance-reduced stochastic optimization for convex optimization.
- **Acceleration by smoothing.**

Axis 2: Deep kernel machines

- **Convolutional kernel networks.**
- Applications to image retrieval and image super-resolution.

Axis 3: Sparse estimation and pluri-disciplinary research

- Complexity analysis of the Lasso regularization path.

Contributions of this HdR

Axis 1: large-scale optimization for machine learning

- Structured MM algorithms for structured problems.
- Variance-reduced stochastic optimization for convex optimization.
- **Acceleration by smoothing.**

Axis 2: Deep kernel machines

- **Convolutional kernel networks.**
- Applications to image retrieval and image super-resolution.

Axis 3: Sparse estimation and pluri-disciplinary research

- Complexity analysis of the Lasso regularization path.
- Path selection in graphs and isoform discovery in RNA-Seq data.

Contributions of this HdR

Axis 1: large-scale optimization for machine learning

- Structured MM algorithms for structured problems.
- Variance-reduced stochastic optimization for convex optimization.
- **Acceleration by smoothing.**

Axis 2: Deep kernel machines

- **Convolutional kernel networks.**
- Applications to image retrieval and image super-resolution.

Axis 3: Sparse estimation and pluri-disciplinary research

- Complexity analysis of the Lasso regularization path.
- Path selection in graphs and isoform discovery in RNA-Seq data.
- A computational model for V4 in neuroscience.

Part III: Focus on acceleration techniques for machine learning

Focus on acceleration techniques for machine learning

Part of the PhD thesis of Honghzou Lin (defense on Nov. 16th).

Publications and pre-prints

H. Lin, J. Mairal and Z. Harchaoui. A Generic Quasi-Newton Algorithm for Faster Gradient-Based Optimization. *arXiv:1610.00960.* 2017

H. Lin, J. Mairal and Z. Harchaoui. A Universal Catalyst for First-Order Optimization. *Adv. NIPS* 2015.

Focus on acceleration techniques for machine learning

Minimizing large finite sums

Consider the minimization of a large sum of convex functions

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x) + \psi(x) \right\},$$

where each f_i is **smooth and convex** and ψ is a convex regularization penalty but not necessarily differentiable.

Goal of this work

- Design accelerated methods for minimizing **large finite sums**.
- Give a **generic acceleration schemes** which can be applied to previously un-accelerated algorithms.

Focus on acceleration techniques for machine learning

Minimizing large finite sums

Consider the minimization of a large sum of convex functions

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x) + \psi(x) \right\},$$

where each f_i is **smooth and convex** and ψ is a convex regularization penalty but not necessarily differentiable.

Goal of this work

- Design accelerated methods for minimizing **large finite sums**.
- Give a **generic acceleration schemes** which can be applied to previously un-accelerated algorithms.

Two solutions: (1) Catalyst (Nesterov's acceleration);

Focus on acceleration techniques for machine learning

Minimizing large finite sums

Consider the minimization of a large sum of convex functions

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x) + \psi(x) \right\},$$

where each f_i is **smooth and convex** and ψ is a convex regularization penalty but not necessarily differentiable.

Goal of this work

- Design accelerated methods for minimizing **large finite sums**.
- Give a **generic acceleration schemes** which can be applied to previously un-accelerated algorithms.

Two solutions: (2) QuickeNing (Quasi Newton);

Focus on acceleration techniques for machine learning

Parenthesis: Consider the minimization of a μ-strongly convex and L-smooth function with a first-order method.

$$\min_{x \in \mathbb{R}^p} f(x).$$

The gradient descent method:

$$x_t \leftarrow x_{t-1} - \frac{1}{L}\nabla f(x_{t-1}).$$

- Iteration-complexity to guarantee $f(x_t) - f^\star \leq \varepsilon$:

$$O\left(\frac{L}{\mu}\log\left(\frac{1}{\varepsilon}\right)\right).$$

Focus on acceleration techniques for machine learning

Parenthesis: Consider the minimization of a μ-strongly convex and L-smooth function with a first-order method.

$$\min_{x \in \mathbb{R}^p} f(x).$$

The accelerated gradient descent method [Nesterov, 1983]:

$$x_t \leftarrow y_{t-1} - \frac{1}{L}\nabla f(y_{t-1}) \quad \text{and} \quad y_t = x_t + \beta_t(x_t - x_{t-1}).$$

- Iteration-complexity to guarantee $f(x_t) - f^\star \leq \varepsilon$:

$$O\left(\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\varepsilon}\right)\right).$$

- Works often in practice, even though the analysis is a worst case.

Focus on acceleration techniques for machine learning

Parenthesis: Consider the minimization of a μ-strongly convex and L-smooth function with a first-order method.

$$\min_{x \in \mathbb{R}^p} f(x).$$

Limited memory Quasi Newton (L-BFGS):

$$x_t \leftarrow x_{t-1} - \eta_t H_t \nabla f(x_{t-1}) \quad \text{with} \quad H_t \approx (\nabla^2 f(x_{t-1}))^{-1}.$$

- L-BFGS uses implicitly a low-rank matrix H_t.
- Iteration-complexity to guarantee $f(x_t) - f^\star \leq \varepsilon$ is **no better than gradient descent**.
- **outstanding performance** in practice, when well implemented.

[Nocedal, 1980, Liu and Nocedal, 1989].

Focus on acceleration techniques for machine learning

The Moreau-Yosida smoothing

Given $f : \mathbb{R}^d \to \mathbb{R}$ a convex function, the Moreau-Yosida smoothing of f is the function $F : \mathbb{R}^d \to \mathbb{R}$ defined as

$$F(x) = \min_{w \in \mathbb{R}^d} \left\{ f(w) + \frac{\kappa}{2}\|w - x\|^2 \right\}.$$

The **proximal operator** $p(x)$ is the unique minimizer of the problem.

Focus on acceleration techniques for machine learning

The Moreau-Yosida smoothing

Given $f : \mathbb{R}^d \to \mathbb{R}$ a convex function, the Moreau-Yosida smoothing of f is the function $F : \mathbb{R}^d \to \mathbb{R}$ defined as

$$F(x) = \min_{w \in \mathbb{R}^d} \left\{ f(w) + \frac{\kappa}{2} \|w - x\|^2 \right\}.$$

The **proximal operator** $p(x)$ is the unique minimizer of the problem.

Properties [see Lemaréchal and Sagastizábal, 1997]

- minimizing f and F is equivalent.
- F is κ-smooth (even when f is nonsmooth) and

$$\nabla F(x) = \kappa(x - p(x)).$$

- the condition number of F is $1 + \frac{\kappa}{\mu}$ (when $\mu > 0$).

Focus on acceleration techniques for machine learning

A naive approach consists of **minimizing the smoothed objective F instead of f** with a method designed for smooth optimization.

Consider indeed

$$x_t = x_{t-1} - \frac{1}{\kappa} \nabla F(x_{t-1}).$$

By rewriting the gradient $\nabla F(x_{t-1})$ as $\kappa(x_{t-1} - p(x_{t-1}))$, we obtain

$$x_t = p(x_{t-1}) = \arg\min_{w \in \mathbb{R}^p} \left\{ f(w) + \frac{\kappa}{2} \|w - x_{t-1}\|^2 \right\}.$$

This is exactly the **proximal point algorithm** [Rockafellar, 1976].

Focus on acceleration techniques for machine learning

A naive approach consists of **minimizing the smoothed objective F instead of f** with a method designed for smooth optimization.

Consider indeed

$$x_t = x_{t-1} - \frac{1}{\kappa}\nabla F(x_{t-1}).$$

By rewriting the gradient $\nabla F(x_{t-1})$ as $\kappa(x_{t-1} - p(x_{t-1}))$, we obtain

$$x_t = p(x_{t-1}) = \operatorname*{arg\,min}_{w \in \mathbb{R}^p} \left\{ f(w) + \frac{\kappa}{2}\|w - x_{t-1}\|^2 \right\}.$$

This is exactly the **proximal point algorithm** [Rockafellar, 1976].

Remarks

- we can do better than gradient descent;
- computing $p(x_{t-1})$ has a cost.

Focus on acceleration techniques for machine learning

Catalyst is a particular **accelerated proximal point algorithm with inexact gradients** [Güler, 1992].

$$x_t \approx p(y_{t-1}) \quad \text{and} \quad y_t = x_t + \beta_t(x_t - x_{t-1})$$

The quantity x_t is obtained by using an optimization method for approximately solving:

$$x_t \approx \arg\min_{w \in \mathbb{R}^p} \left\{ f(w) + \frac{\kappa}{2}\|w - y_{t-1}\|^2 \right\},$$

Catalyst provides Nesterov's acceleration to $\mathcal{M}$ with...

- **restart strategies** for solving the sub-problems;
- **global complexity analysis** resulting in theoretical acceleration.
- **parameter choices** (as a consequence of the complexity analysis);

Focus on acceleration techniques for machine learning

QuickeNing uses a similar strategy with L-BFGS.

Main recipe

- L-BFGS applied to the **smoothed objective** F with **inexact gradients**.
- inexact gradients are obtained by **solving sub-problems** using a first-order optimization method $\mathcal{M}$;
- as in Catalyst, one should choose a method $\mathcal{M}$ that is **able to adapt to the problem structure** (finite sum, composite).
- replace L-BFGS steps by proximal point steps if no sufficient decrease is estimated $\Rightarrow$ **no line search on** F;

Focus on acceleration techniques for machine learning

QuickeNing uses a similar strategy with L-BFGS.

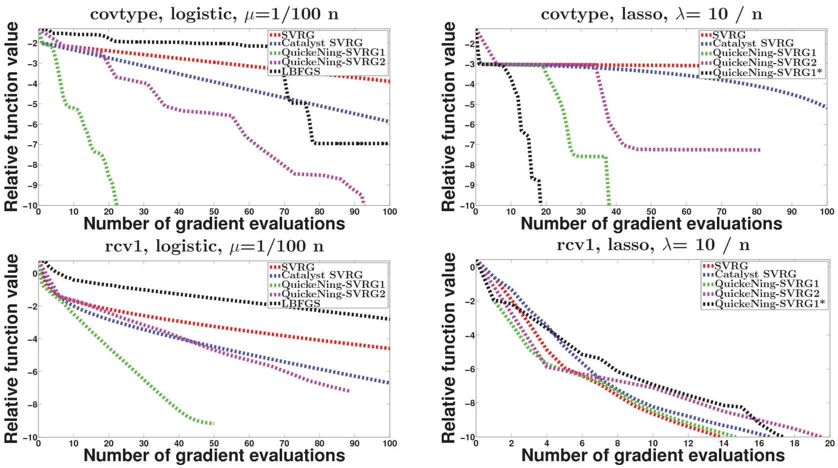
Main recipe

- L-BFGS applied to the **smoothed objective** F with **inexact gradients**.
- inexact gradients are obtained by **solving sub-problems** using a first-order optimization method $\mathcal{M}$;
- as in Catalyst, one should choose a method $\mathcal{M}$ that is **able to adapt to the problem structure** (finite sum, composite).
- replace L-BFGS steps by proximal point steps if no sufficient decrease is estimated $\Rightarrow$ **no line search on** F;

Remark

- often outperform Catalyst in practice (but not in theory).

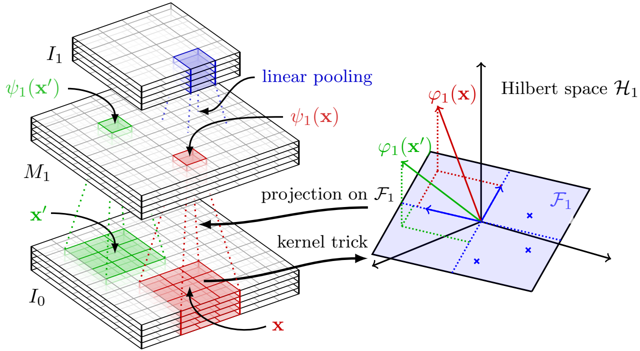
Focus on acceleration techniques for machine learning



- QuickeNing-SVRG $\geq$ SVRG;
- QuickeNing-SVRG $\geq$ Catalyst-SVRG in 10/12 cases.

Part IV: Focus on convolutional kernel networks

Focus on convolutional kernel networks



Publications and pre-prints

A. Bietti and J. Mairal. Invariance and Stability of Deep Convolutional Representations. *Adv. NIPS* 2017.
J. Mairal. End-to-End Kernel Learning with Supervised Convolutional Kernel Networks. *Adv. NIPS* 2016.
J. Mairal, P. Koniusz, Z. Harchaoui and C. Schmid. Convolutional Kernel Networks. *Adv. NIPS* 2014.

Focus on convolutional kernel networks

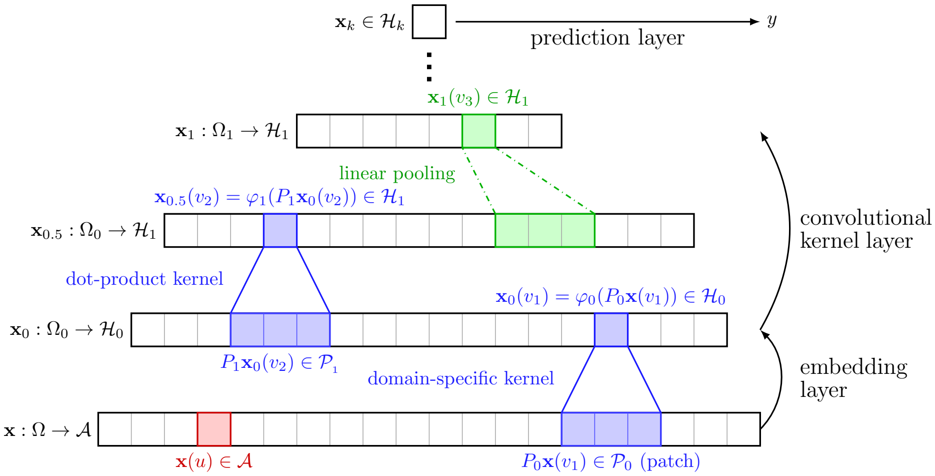


Illustration of multilayer convolutional kernel for 1D discrete signals.

(Figure produced by Dexiong Chen)

Focus on convolutional kernel networks

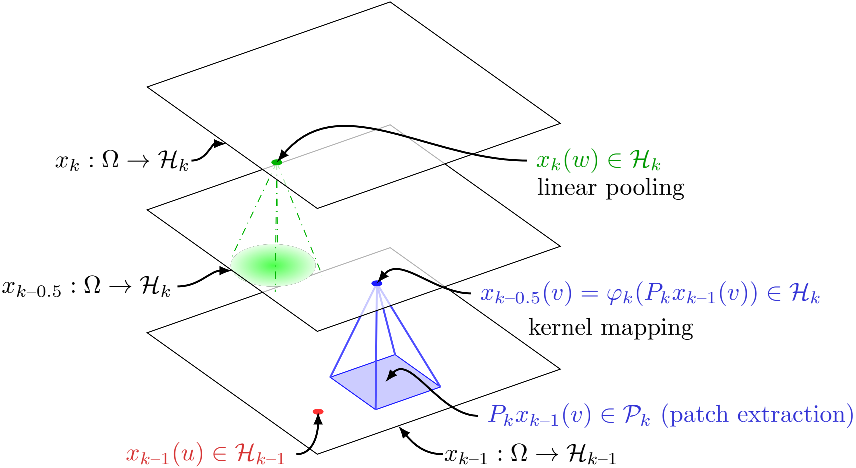
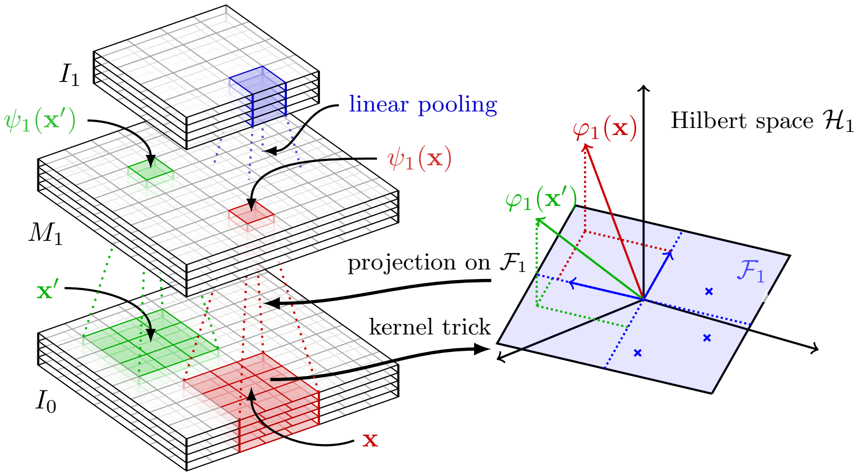


Illustration of multilayer convolutional kernel for 2D continuous signals.

Focus on convolutional kernel networks



Learning mechanism of CKNs between layers 0 and 1.

Focus on convolutional kernel networks

Main principles

- A multilayer kernel, which builds upon similar principles as a convolutional neural net (**multiscale, local stationarity**).

Focus on convolutional kernel networks

Main principles

- A multilayer kernel, which builds upon similar principles as a convolutional neural net (**multiscale, local stationarity**).
- When going up in the hierarchy, we represent **larger neighborhoods with more invariance**;

Focus on convolutional kernel networks

Main principles

- A multilayer kernel, which builds upon similar principles as a convolutional neural net (**multiscale, local stationarity**).
- When going up in the hierarchy, we represent **larger neighborhoods with more invariance**;
- The first layer may encode **domain-specific knowledge**;

Focus on convolutional kernel networks

Main principles

- A multilayer kernel, which builds upon similar principles as a convolutional neural net (**multiscale, local stationarity**).
- When going up in the hierarchy, we represent **larger neighborhoods with more invariance**;
- The first layer may encode **domain-specific knowledge**;
- We build a sequence of functional spaces and data representations that are **decoupled from learning**...

Focus on convolutional kernel networks

Main principles

- A multilayer kernel, which builds upon similar principles as a convolutional neural net (**multiscale, local stationarity**).
- When going up in the hierarchy, we represent **larger neighborhoods with more invariance**;
- The first layer may encode **domain-specific knowledge**;
- We build a sequence of functional spaces and data representations that are **decoupled from learning**...
- But, we learn **linear subspaces** in RKHSs, where we project data, providing a new type of CNN with a **geometric interpretation**.

Focus on convolutional kernel networks

Main principles

- A multilayer kernel, which builds upon similar principles as a convolutional neural net (**multiscale, local stationarity**).
- When going up in the hierarchy, we represent **larger neighborhoods with more invariance**;
- The first layer may encode **domain-specific knowledge**;
- We build a sequence of functional spaces and data representations that are **decoupled from learning**...
- But, we learn **linear subspaces** in RKHSs, where we project data, providing a new type of CNN with a **geometric interpretation**.
- Learning may be **unsupervised** (reduce approximation error) or **supervised** (via backpropagation).

Focus on convolutional kernel networks

Remarks - In practice

- extremely simple to use in unsupervised setting. Is it easier to use than regular CNNs for supervised learning?
- competitive results for various tasks (super-resolution, retrieval,. . .).

Focus on convolutional kernel networks

Remarks - In practice

- extremely simple to use in unsupervised setting. Is it easier to use than regular CNNs for supervised learning?
- competitive results for various tasks (super-resolution, retrieval,...).

Remarks - In theory [Bietti and Mairal, 2017]

- **invariance and stability to deformations**.
- may encode invariance to various **groups of transformations**.
- The kernel representation does not lose **signal information**.
- Our RKHSs contain **classical CNNs** with homogeneous activation functions. Can we say something about them?

[Mallat, 2012]

Focus on convolutional kernel networks

Figure: Results for x3 upscaling.

Focus on convolutional kernel networks

Figure: Bicubic.

Focus on convolutional kernel networks

Figure: SCKN.

Part V: Conclusion and perspectives

Main perspectives

Beyond the challenges already raised for each paradigm, which remain unsolved in large parts, here is a selection of three perspectives.

on optimization

- go beyond the ERM formulation. Develop algorithms for Nash equilibriums, saddle-point problems, active learning. . .

on deep kernel machines

- work with structured data (sequences, graphs...) and develop pluri-disciplinary collaborations.

on sparsity

- simplicity, stability, and compositional principles are needed for unsupervised learning, but where?

References I

Alberto Bietti and Julien Mairal. Group invariance and stability to deformations of deep convolutional representations. *arXiv preprint arXiv:1706.03078*, 2017.

Léon Bottou, Jonas Peters, Joaquin Quiñonero-Candela, Denis X Charles, D Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard, and Ed Snelson. Counterfactual reasoning and learning systems: The example of computational advertising. *The Journal of Machine Learning Research*, 14 (1):3207–3260, 2013.

Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *arXiv preprint arXiv:1606.04838*, 2016.

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61, 1999.

D. Ciresan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

References II

David Corfield, Bernhard Schölkopf, and Vladimir Vapnik. Falsificationism and statistical learning theory: Comparing the popper and vapnik-chervonenkis dimensions. *Journal for General Philosophy of Science*, 40(1):51–58, 2009.

O. Güler. New proximal point algorithms for convex minimization. *SIAM Journal on Optimization*, 2(4):649–664, 1992.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *P. IEEE*, 86(11):2278–2324, 1998.

Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

Claude Lemaréchal and Claudia Sagastizábal. Practical aspects of the moreau–yosida regularization: Theoretical preliminaries. *SIAM Journal on Optimization*, 7(2):367–385, 1997.

References III

D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012.

K-R Müller, Sebastian Mika, Gunnar Ratsch, Koji Tsuda, and Bernhard Scholkopf. An introduction to kernel-based learning algorithms. *IEEE transactions on neural networks*, 12(2):181–201, 2001.

Yurii Nesterov. A method for unconstrained convex minimization problem with the rate of convergence o (1/k2). In *Doklady an SSSR*, volume 269, pages 543–547, 1983.

Jorge Nocedal. Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, 35(151):773–782, 1980.

B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381: 607–609, 1996.

R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14(5):877–898, 1976.

References IV

Bernhard Schölkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. *arXiv preprint arXiv:1206.6471*, 2012.

John Shawe-Taylor and Nello Cristianini. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2004.

R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society: Series B*, 58(1):267–288, 1996.

Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1995.

D. Wrinch and H. Jeffreys. XLII. On certain fundamental principles of scientific inquiry. *Philosophical Magazine Series 6*, 42(249):369–390, 1921.